



ONLINE TICKET RESERVATION SYSTEM





BOOTCAMP



Sabareesh M – 18EUIT128

Nishit M – 18EUCS152

Subash Niranjana B C – 18EUEC160

Kanisha R – 18EUCS053

Makeshkumar R – 18EUCS069

DAY 2 SUBMISSION

LOGIN PAGE CODE:

```
package bootathon;

import java.util.regex.*;
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionListener;
import java.awt.event.ItemEvent;
import java.awt.event.ItemListener;
import java.awt.event.ActionEvent;
import java.sql.DriverManager;
import java.sql.*;

public class LoginPage extends JFrame{

    Container c=getContentPane();

    public int user_id=1000;
    int count=0;

    JPanel panel=new JPanel();

    Image icon = Toolkit.getDefaultToolkit().getImage("F:\\\\Bootathon\\\\railway.png");

    JLabel headingLabel=new JLabel();
    JLabel usertypeLabel=new JLabel();
```

```

JLabel usernameLabel=new JLabel();
JLabel passwordLabel=new JLabel();
JLabel nothavingLabel=new JLabel();

JTextField usernameField=new JTextField();

JPasswordField passwordField=new JPasswordField();

String usertype[]={"User","Admin"};
JComboBox usertypeCombobox=new JComboBox(usertype);

JButton signinButton=new JButton();
JButton signupButton=new JButton();
JButton exitButton=new JButton();

public LoginPage() {
    setLayoutManager();
    setNavigationPanel();
    setLocationAndSize();
    addCompenentsToTheConatiner();
    addItemEvent();
    addActionEvent();
    frame();
}

public void frame() {
    setTitle("Login");
    setSize(750,600);
    setResizable(false);
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setIconImage(icon);
}

```

```

        setLocationRelativeTo(null);

        setVisible(true);
    }

    public void setLayoutManager() {
        c.setLayout(null);
    }

    public void setNavigationPanel() {
        panel.setBackground(new Color(65, 105, 225));
        panel.setBounds(0, 0, 750, 80);
        panel.add(headingLabel);
    }

    public void addItemEvent() {
        usertypeCombobox.addItemListener(new ItemListener() {

            @Override
            public void itemStateChanged(ItemEvent e) {
                if(usertypeCombobox.getSelectedItem()=="Admin") {
                    signupButton.setEnabled(false);
                }
                if(usertypeCombobox.getSelectedItem()=="User") {
                    signupButton.setEnabled(true);
                }
            }
        });
    }

    public void setLocationAndSize() {

        headingLabel.setBounds(90, 0, 550, 80);
    }

```

```
headingLabel.setText("Railway Reservation System");

headingLabel.setHorizontalAlignment(JLabel.CENTER);

headingLabel.setFont(new Font("Times Newroman", Font.BOLD, 20));

headingLabel.setForeground(Color.white);


usertypeLabel.setBounds(235, 150, 100, 50);

usertypeLabel.setText("Ustertype");

usertypeLabel.setFont(new Font("Times Newroman", Font.BOLD, 18));


usernameLabel.setBounds(235, 250, 100, 50);

usernameLabel.setText("Username");

usernameLabel.setFont(new Font("Times Newroman", Font.BOLD, 18));


passwordLabel.setBounds(235, 350, 100, 50);

passwordLabel.setText("Password");

passwordLabel.setFont(new Font("Times Newroman", Font.BOLD, 18));


usernameField.setBounds(360, 262, 150, 30);


passwordField.setBounds(360, 362, 150, 30);


usertypeCombobox.setBounds(360, 162, 150, 30);


signinButton.setBounds(210, 462, 100, 30);

signinButton.setText("Sign in");

signinButton.setBackground(new Color(65, 105, 225));

signinButton.setForeground(Color.white);


signupButton.setBounds(330, 462, 100, 30);
```

```

        signupButton.setText("Sign up");
        signupButton.setBackground(new Color(65, 105, 225));
        signupButton.setForeground(Color.white);

        exitButton.setBounds(450, 462, 100, 30);
        exitButton.setText("Exit");
        exitButton.setBackground(new Color(65, 105, 225));
        exitButton.setForeground(Color.white);

    }

    public void addCompenentsToTheConatiner() {
        c.add(panel);
        c.setBackground(new Color(135, 206, 250));
        c.add(usertypeLabel);
        c.add(usernameLabel);
        c.add(passwordLabel);
        c.add(usernameField);
        c.add(passwordField);
        c.add(usertypeCombobox);
        c.add(signinButton);
        c.add(signupButton);
        c.add(exitButton);

    }

    public void addActionEvent() {
        signupButton.addActionListener(new ActionListener() {
            @Override

```

```

        public void actionPerformed(ActionEvent e) {
            try {
                dispose();
                new SignupPage();
            }
            catch (Exception ex) {
                ex.printStackTrace();
            }
        }
    });

    signinButton.addActionListener(new ActionListener() {
        @Override
        public void actionPerformed(ActionEvent e) {
            try{
                signin();
            }
            catch(Exception ex){
                ex.printStackTrace();
            }
        }
    });

    exitButton.addActionListener(new ActionListener() {
        @Override
        public void actionPerformed(ActionEvent e) {

            System.exit(0);

        }
    });
}

```



```

public void signin() throws Exception{
    if(usertypeCombobox.getSelectedItem()=="User") {
        try{
            int flag=0;
            Class.forName("com.mysql.cj.jdbc.Driver");
            Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/reservation?serverTimezone=UTC","root",
,"");

            Statement stmt=con.createStatement();
            String user_name=usernameField.getText();
            String pass_word=passwordField.getText();
            String query="Select * from userid";
            ResultSet rs=stmt.executeQuery(query);
            while(rs.next()){
                if(rs.getString("username").equals(user_name) &&
rs.getString("password").equals(pass_word)) {
                    flag=1;
                    break;
                }
            }
            if(flag==1){
                System.out.println("Next page opens");//constructor should be
called
                dispose();
            }
            if(flag==0){
                JOptionPane.showMessageDialog(new JFrame(),"Check the username
and password or Please create an account ", " Account not found ",JOptionPane.WARNING_MESSAGE);
                dispose();
                new SignupPage();
            }
        }
    }
}

```

```

        }

        con.close();

    }

    catch(Exception e){

        e.printStackTrace();

    }

}

else {

    try{

        int flag=0;

        Class.forName("com.mysql.cj.jdbc.Driver");

        Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/reservation?serverTimezone=UTC","root"
,"");

        Statement stmt=con.createStatement();

        String user_name=usernameField.getText();

        String pass_word=passwordField.getText();

        String query="Select * from admin";

        ResultSet rs=stmt.executeQuery(query);

        while(rs.next()){

            if(rs.getString("username").equals(user_name) &&
rs.getString("password").equals(pass_word)) {

                flag=1;

                break;

            }

        }

        if(flag==1){

            System.out.println("Admin page opens");//constructor should be
called

            dispose();

        }

        if(flag==0){

```

```

        JOptionPane.showMessageDialog(new JFrame(), "Make sure the
username and password is correct", " Access Denied", JOptionPane.ERROR_MESSAGE);

        dispose();

        new LoginPage();

    }

    con.close();

}

catch(Exception e){

    e.printStackTrace();

}

}

}

}

public static void main(String[] args) {

    new LoginPage();

}

}

}

```

SIGN UP PAGE CODE:

```

package bootathon;

import java.util.regex.*;

import javax.swing.*;

import java.awt.*;

import java.awt.event.ActionListener;

import java.awt.event.ItemEvent;

import java.awt.event.ItemListener;

import java.awt.event.ActionEvent;

import java.sql.DriverManager;

```

```
import java.sql.*;

public class SignupPage extends JFrame{

    Container c=getContentPane();

    JPanel panel=new JPanel();

    JLabel label_firstName=new JLabel();
    JLabel label_lastName=new JLabel();
    JLabel label_age=new JLabel();
    JLabel label_gender=new JLabel();
    JLabel label_phonenum=new JLabel();
    JLabel label_username=new JLabel();
    JLabel label_password=new JLabel();
    JLabel label_male=new JLabel();
    JLabel label_female=new JLabel();

    JTextField textfield_firstName=new JTextField();
    JTextField textfield_lastName=new JTextField();
    JTextField textfield_age=new JTextField();
    JTextField textfield_phonenum=new JTextField();
    JTextField textfield_username=new JTextField();

    JPasswordField passwordfield_password=new JPasswordField();

    JRadioButton radiobutton_male=new JRadioButton();
    JRadioButton radiobutton_female=new JRadioButton();

    Image icon = Toolkit.getDefaultToolkit().getImage("F:\\Bootathon\\railway.png");
```

```

JLabel headingLabel=new JLabel();

JButton signupButton=new JButton();

JButton exitButton=new JButton();

JButton backButton=new JButton();


public SignupPage() {
    setLayoutManager();
    setNavigationPanel();
    setLocationAndSize();
    addCompenentsToTheConatiner();
//    addItemEvent();
    addActionEvent();
    buttonGroup();
    frame();
}

public void buttonGroup() {
    ButtonGroup bg=new ButtonGroup();
    bg.add(radiobutton_male);
    bg.add(radiobutton_female);
}

public void frame() {
    setTitle("Signup");
    setSize(750,600);
    setResizable(false);
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setIconImage(icon);
}

```

```

        setLocationRelativeTo(null);

        setVisible(true);
    }

    public void setLayoutManager() {

        c.setLayout(null);

    }

    public void setNavigationPanel() {

        panel.setBackground(new Color(65, 105, 225));

        panel.setBounds(0, 0, 750, 80);

        panel.add(headingLabel);

    }


    public void setLocationAndSize() {

        headingLabel.setBounds(90, 0, 550, 80);

        headingLabel.setText("Create an Account");

        headingLabel.setHorizontalAlignment(JLabel.CENTER);

        headingLabel.setFont(new Font("Times Newroman", Font.BOLD, 20));

        headingLabel.setForeground(Color.white);


        label_firstName.setBounds(235, 100, 100, 50);

        label_firstName.setText("First Name");

        label_firstName.setFont(new Font("Times Newroman", Font.BOLD, 15));

        label_firstName.setForeground(Color.black);


        label_lastName.setBounds(235, 150, 100, 50);

        label_lastName.setText("Last Name");

        label_lastName.setFont(new Font("Times Newroman", Font.BOLD, 15));

        label_lastName.setForeground(Color.black);
    }

```

```
label_age.setBounds(235, 200, 100, 50);  
label_age.setText("Age");  
label_age.setFont(new Font("Times Newroman", Font.BOLD, 15));  
label_age.setForeground(Color.black);  
  
label_gender.setBounds(235, 250, 100, 50);  
label_gender.setText("Gender");  
label_gender.setFont(new Font("Times Newroman", Font.BOLD, 15));  
label_gender.setForeground(Color.black);  
  
label_phonenum.setBounds(235, 300, 150, 50);  
label_phonenum.setText("Phone number");  
label_phonenum.setFont(new Font("Times Newroman", Font.BOLD, 15));  
label_phonenum.setForeground(Color.black);  
  
label_username.setBounds(235, 350, 150, 50);  
label_username.setText("Username");  
label_username.setFont(new Font("Times Newroman", Font.BOLD, 15));  
label_username.setForeground(Color.black);  
  
label_password.setBounds(235, 400, 150, 50);  
label_password.setText("Password");  
label_password.setFont(new Font("Times Newroman", Font.BOLD, 15));  
label_password.setForeground(Color.black);  
  
label_male.setBounds(380, 250, 150, 50);  
label_male.setText("male");  
label_male.setFont(new Font("Times Newroman", Font.BOLD, 15));  
label_male.setForeground(Color.black);  
  
label_female.setBounds(460, 250, 150, 50);
```

```
label_female.setText("female");  
label_female.setFont(new Font("Times Newroman", Font.BOLD, 15));  
label_female.setForeground(Color.black);
```

```
textfield_firstName.setBounds(360, 112, 150, 30);  
textfield_lastName.setBounds(360, 162, 150, 30);  
textfield_age.setBounds(360, 212, 150, 30);  
textfield_phonenum.setBounds(360, 312, 150, 30);  
textfield_username.setBounds(360, 362, 150, 30);
```

```
passwordfield_password.setBounds(360, 412, 150, 30);
```

```
radiobutton_male.setBounds(360, 265, 20, 20);  
radiobutton_male.setBackground(new Color(135, 206, 250));
```

```
radiobutton_female.setBounds(440, 265, 20, 20);  
radiobutton_female.setBackground(new Color(135, 206, 250));  
radiobutton_male.setSelected(true);
```

```
signupButton.setBounds(200, 482, 100, 30);  
signupButton.setText("Sign up");  
signupButton.setBackground(new Color(65, 105, 225));  
signupButton.setForeground(Color.white);
```

```
exitButton.setBounds(500, 482, 100, 30);  
exitButton.setText("Exit");  
exitButton.setBackground(new Color(65, 105, 225));  
exitButton.setForeground(Color.white);
```

```
backbutton.setBounds(350, 482, 100, 30);
```



```
        backButton.setText("Back");

        backButton.setBackground(new Color(65, 105, 225));

        backButton.setForeground(Color.white);

    }

    public void addCompenentsToTheConatiner() {

        c.add(panel);

        c.setBackground(new Color(135, 206, 250));

        c.add(label_firstName);
        c.add(label_lastName);
        c.add(label_age);
        c.add(label_gender);
        c.add(label_phonenum);
        c.add(label_username);
        c.add(label_password);
        c.add(label_male);
        c.add(label_female);

        c.add(signupButton);
        c.add(exitButton);
        c.add(backButton);

        c.add(textfield_firstName);
        c.add(textfield_lastName);
        c.add(textfield_age);
        c.add(textfield_phonenum);
        c.add(textfield_username);
```

```

        c.add(passwordfield_password);

        c.add(radiobutton_male);
        c.add(radiobutton_female);

    }

    public void addActionEvent() {
        signupButton.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                try {
                    signup();
                }
                catch (Exception ex) {
                    ex.printStackTrace();
                }
            }
        });

        exitButton.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {

                System.exit(0);
            }
        });

        backButton.addActionListener(new ActionListener() {

```

```

        public void actionPerformed(ActionEvent e) {
            dispose();
            new LoginPage();
        }
    });
}

```

```

public void signup() throws Exception{
    try{
        int flag=0;
        Class.forName("com.mysql.cj.jdbc.Driver");
        Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/reservation?serverTimezone=UTC","root",
"");

        Statement stmt=con.createStatement();
        Statement stmt1=con.createStatement();
        String first_name=textfield_firstName.getText();
        String last_name=textfield_lastName.getText();
        String s_age=textfield_age.getText();
        String gender;
        String phone_num=textfield_phonenum.getText();
        String user_name=textfield_username.getText();
        String pass_word=passwordfield_password.getText();
        if(radiobutton_male.isSelected()==true) {
            gender="male";
        }
        else {

```

```

        gender="female";
    }

    Boolean a=Pattern.matches("[a-zA-Z]*${1,20}", first_name);
    Boolean b=Pattern.matches("[a-zA-Z]*${1,20}", last_name);
    Boolean d=Pattern.matches("\\d{10}$", phone_num);
    Boolean e=Pattern.matches("[a-zA-Z]\\w{5,20}$", user_name);

    String query="SELECT * from userid";
    ResultSet rs=stmt.executeQuery(query);

    while(rs.next()) {
        if(rs.getString("username").equals(user_name)) {
            JOptionPane.showMessageDialog(new JFrame(), "Username not
vaild", "Username already exists!", JOptionPane.ERROR_MESSAGE);

            flag=1;
            con.close();
            break;
        }
    }

    if(flag==0) {
        if(first_name.isEmpty()) {
            JOptionPane.showMessageDialog(new JFrame(), "Firstname cannot
be empty", "INVALID FIRSTNAME", JOptionPane.ERROR_MESSAGE);

            flag=1;
        }

        else if(a==false) {
            JOptionPane.showMessageDialog(new JFrame(), "Firstname should
contain only characters", "INVALID FIRSTNAME", JOptionPane.ERROR_MESSAGE);

            flag=1;
        }

        else if(last_name.isEmpty()) {
            JOptionPane.showMessageDialog(new JFrame(), "Lastname cannot
be empty", "INVALID LASTNAME", JOptionPane.ERROR_MESSAGE);

            flag=1;
        }
    }
}

```

```

    }

    else if(b==false) {

        JOptionPane.showMessageDialog(new JFrame(), "Lastname should
contain only characters", "INVALID LASTNAME", JOptionPane.ERROR_MESSAGE);

        flag=1;

    }

    else if(s_age.isEmpty()){

        JOptionPane.showMessageDialog(new JFrame(), "Age should not be
empty", "AGE IS EMPTY", JOptionPane.YES_OPTION);

        flag=1;

    }

    else if(Integer.parseInt(s_age)<18) {

        JOptionPane.showMessageDialog(new JFrame(), "Age should be
greater than 18", "AGE IS MINIMUM", JOptionPane.OK_OPTION);

        flag=1;

    }

    else if(phone_num.isEmpty()) {

        JOptionPane.showMessageDialog(new JFrame(), "Phonenumber
cannot be empty", "INVALID PHONE NUMBER", JOptionPane.ERROR_MESSAGE);

        flag=1;

    }

    else if(d==false) {

        JOptionPane.showMessageDialog(new JFrame(), "Phone must be 10-
digit number", "INVALID PHONE NUMBER", JOptionPane.ERROR_MESSAGE);

        flag=1;

    }

    else if(user_name.isEmpty()) {

        JOptionPane.showMessageDialog(new JFrame(), "Username cannot
be empty", "INVALID USERNAME", JOptionPane.ERROR_MESSAGE);

        flag=1;

    }

    else if(e==false) {

        JOptionPane.showMessageDialog(new JFrame(), "Username must be

```

```

5-20 characters", "INVALID USERNAME", JOptionPane.ERROR_MESSAGE);

        flag=1;

    }

    else if(pass_word.isEmpty()) {

        JOptionPane.showMessageDialog(new JFrame(), "Password cannot
be empty", "INVALID PASSWORD", JOptionPane.ERROR_MESSAGE);

        flag=1;

    }

    else if(pass_word.length()<8 || pass_word.length()>20) {

        JOptionPane.showMessageDialog(new JFrame(), "Password must
contain 8-20 characters", "INVALID PASSWORD", JOptionPane.ERROR_MESSAGE);

        flag=1;

    }

    if(flag==0) {

        String query1="INSERT into
userid(firstname, lastname, age, gender, phonenumber, username, password)
values('"+first_name+"', '"+last_name+"', '"+Integer.parseInt(s_age)+"', '"+gender+"', '"+phone_num
+'', '"+user_name+"', '"+pass_word+"')";

        stmt1.executeUpdate(query1);

        JOptionPane.showMessageDialog(new JFrame(), "Account created
successfully", "ACCOUNT CREATED", JOptionPane.INFORMATION_MESSAGE);

        JOptionPane.showMessageDialog(new JFrame(), "Please remember
your username and password", "ACCOUNT CREATED", JOptionPane.WARNING_MESSAGE);

        dispose();

        new LoginPage();

    }

    con.close();

    //

}

catch(Exception e) {

```

```

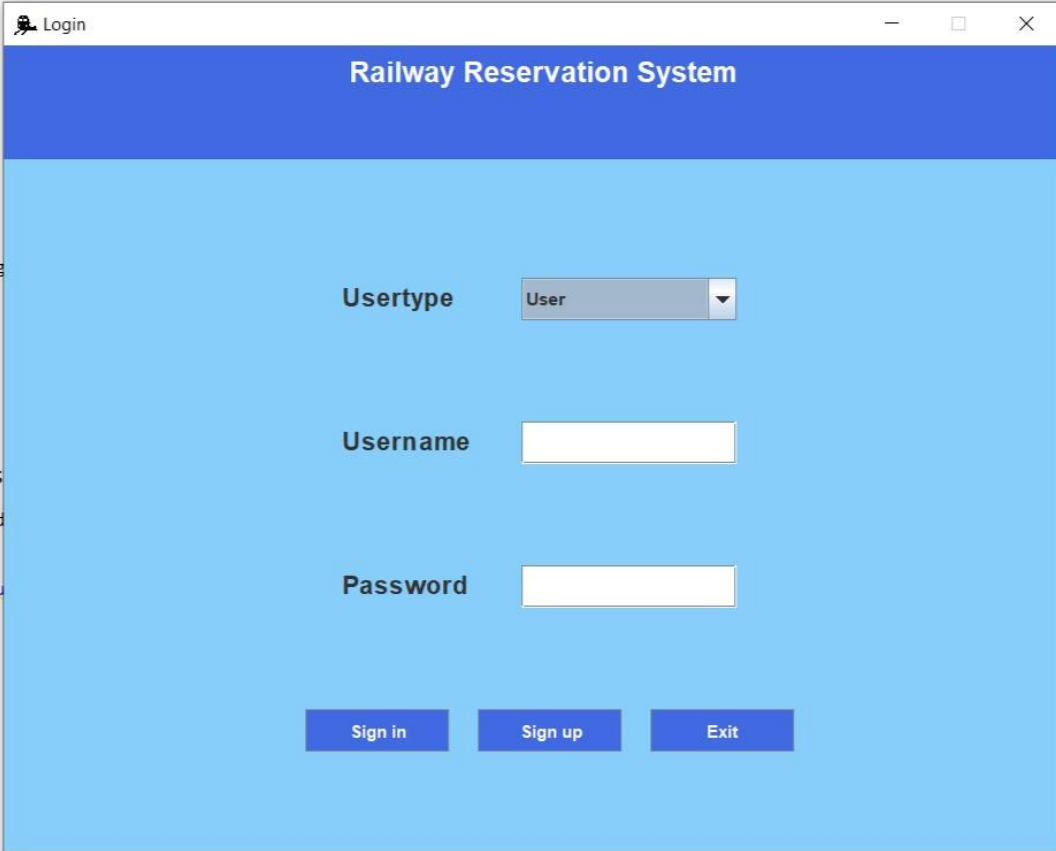
        e.printStackTrace();

        //System.out.print(e);
    }
}

public static void main(String[] args) {
    new SignupPage();
}
}

```

SCREEN SHOTS



The screenshot shows a Java Swing window titled "Login" for the "Railway Reservation System". The window has a blue header bar with the system name. The main area is light blue and contains three input fields: "Usertype" (a dropdown menu currently set to "User"), "Username" (a text box), and "Password" (a text box). At the bottom, there are three blue buttons labeled "Sign in", "Sign up", and "Exit".

Login

Railway Reservation System

Usertype

Username

Password

Account not found

⚠ Check the username and password or Please create an account

Signup

Create an Account

First Name

Last Name

Age

Gender ☒ male ☐ female

Phone number

Username

Password

Signup

Create an Account

First Name

Last Name

Age


Gender

Phone Number

Username

Password

INVALID FIRSTNAME

 Firstname cannot be empty

Signup

Create an Account

First Name

Last Name

Age


Gender

Phone Number

Username

Password

INVALID FIRSTNAME

 Firstname should contain only characters

Signup

Create an Account

First Name

Last Name

Age


Gender

Phone Number

Username

Password

AGE IS MINIMUM

 Age should be greater than 18

Signup

Create an Account

First Name

Last Name

Age


Gender

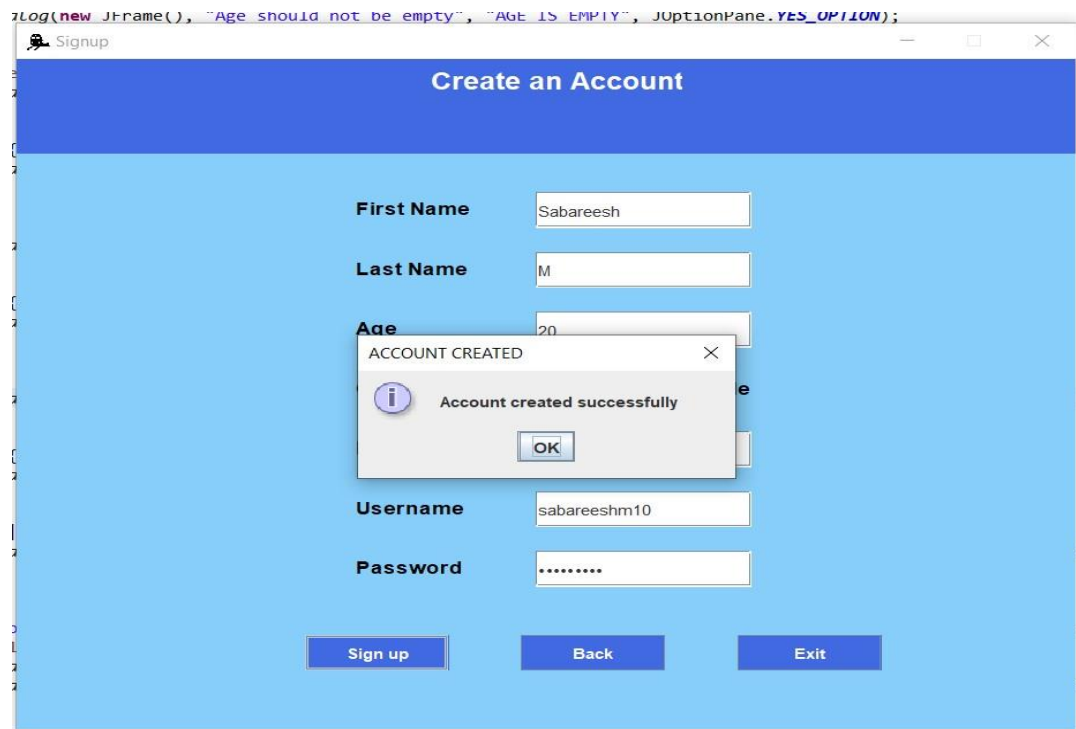
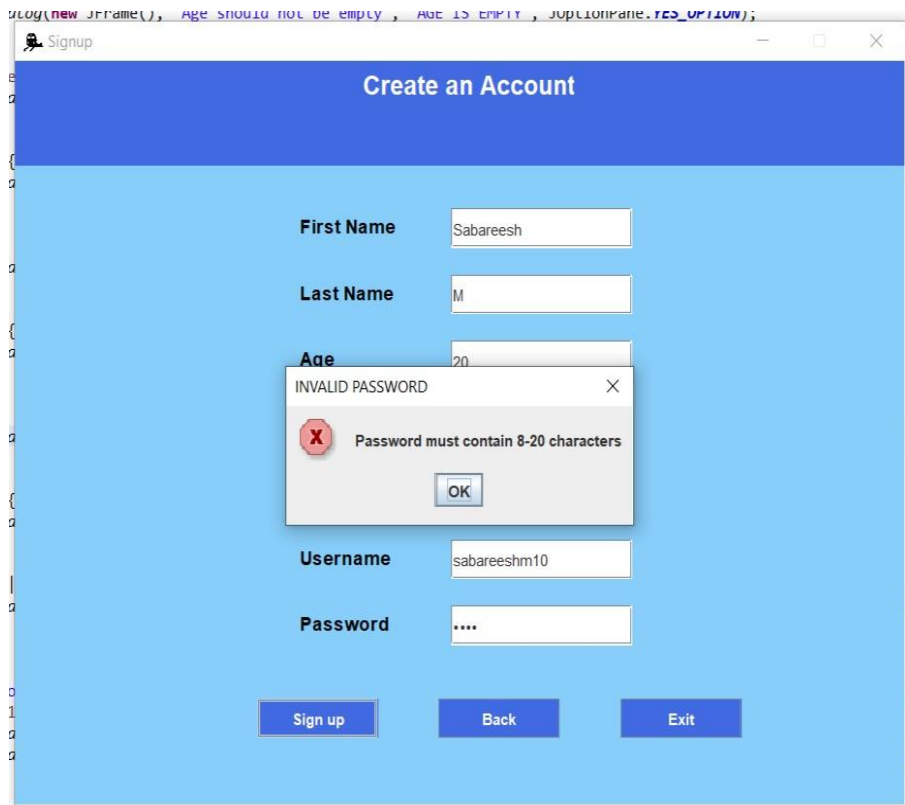
Phone number

Username

Password

INVALID PHONE NUMBER

 Phone must be 10-digit number



```
alog(new JFrame(), "Age should not be empty", "AGE IS EMPTY", JOptionPane.YES_OPTION);
```

Create an Account

First Name: Sabareesh

Last Name: M

Age: 20

Username: sabareeshm10

Password:

ACCOUNT CREATED

Please remember your username and password

OK

Sign up Back Exit

```
alog(new JFrame(), "Age should not be empty", "AGE IS EMPTY", JOptionPane.YES_OPTION);
```

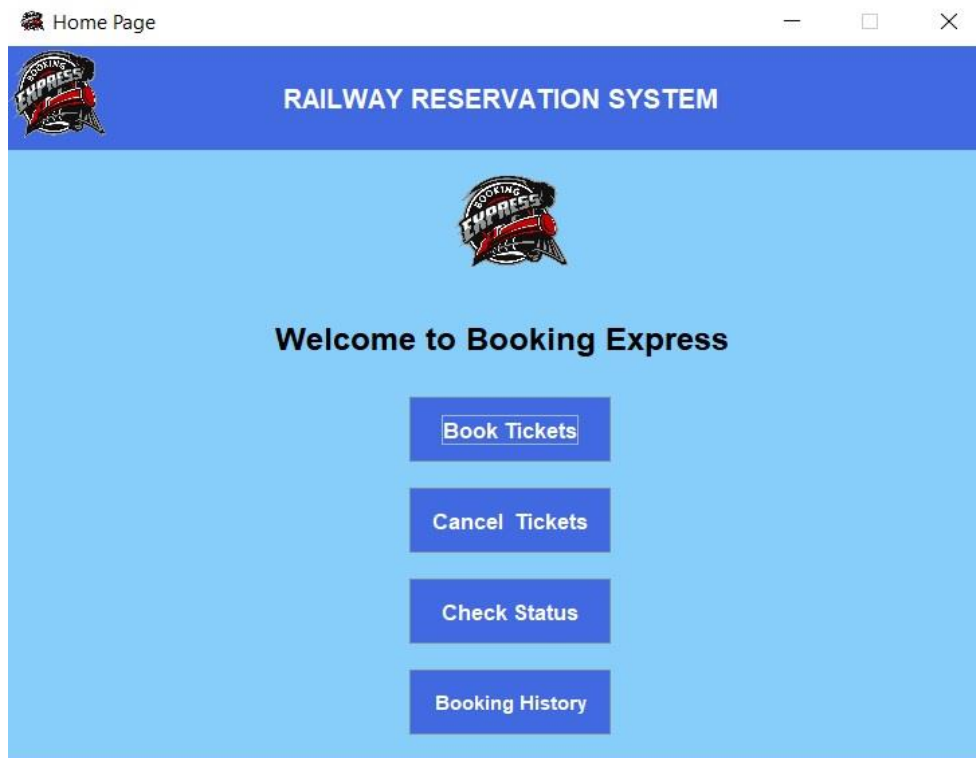
Railway Reservation System

Usertype: User

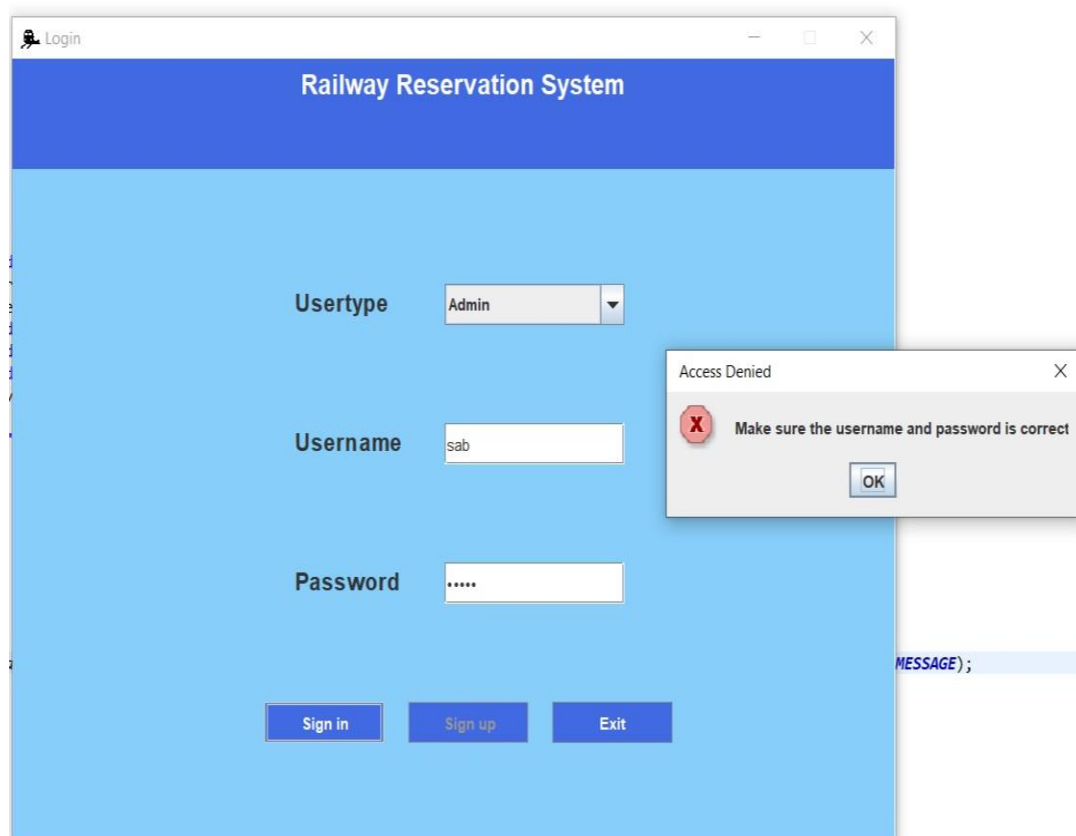
Username: sabareeshm10

Password:

Sign in Sign up Exit



ADMIN PORTAL



Login

Railway Reservation System

Usertype

Username

Password

[Sign in](#) [Sign up](#) [Exit](#)

admin options

RAILWAY RESERVATION SYSTEM

[ADD NEW TRAIN DETAILS](#)

[DELETE TRAIN DETAILS](#)

[EDIT TRAIN DETAILS](#)

[VIEW USER DETAILS](#)